УДК 004.942

О.М. Трофимчук, О.О. Кряжич
Інститут телекомунікацій і глобального інформаційного простору НАН України, Україна
Україна, 03186, м. Київ-186, Чоколівський бульвар, 13

АЛГОРИТМ ОПИСУ ЯРУЖНИХ ЦІЛЬОВИХ ФУНКЦІЙ

O.M. Trofymchuk, O.O. Kryazhych
Institute of Telecommunications and Global Information Space of NAS of Ukraine, Ukraine
Ukraine, 03186, Kyiv-186, 13 Chokolovsky Blvd.

ALGORITHM DESCRIPTIONS GULLY OBJECTIVE FUNCTIONS

А.Н. Трофимчук, О.А. Кряжич
Институт телекоммуникаций и глобального информационного пространства НАН Украины, Украина
Украина, 03186, г. Киев-186, Чоколовский бульвар, 13

АЛГОРИТМ ОПИСАНИЯ ОВРАЖНЫХ ЦЕЛЕВЫХ ФУНКЦИЙ

Представлений алгоритм опису яружних цільових функцій, який пропонується використовувати для детальної візуалізації розповсюдження зон ураження сильнодіючих отруйних речовин у разі виникнення техногенної аварії. Така візуалізація є необхідною і можливою при застосуванні геоінформаційних технологій для оперативного реагування на подію. Розглянуті можливості апроксимації складних функцій за методом Дж. Зойтендейка. При побудові алгоритму застосований метод можливих напрямків до вирішення задач чебишевського наближення з додатковими обмеженнями.
Ключові слова: апроксимація, алгоритм, функція, обмеження, вектор.

The algorithm descriptions gully objective functions. This algorithm can be used for detailed visualization of the distribution of zones of defeat of highly toxic substances in the event of industrial disaster. This rendering is required when using geoinformation technologies for rapid response to an event. Considered the possibility of approximation of complex functions by the method's G. Zoutendijk. When creating the algorithm used method possible directions in problem solving Chebyshev's approximation restrictions.
Key words: approximation, algorithm, function, constraint, vector.

Представлен алгоритм описания овражных целевых функций, который предлагается использовать для детальной визуализации распространения зон поражения сильнодействующих отравляющих веществ в случае возникновения техногенной аварии. Такая визуализация необходима и возможна при использовании геоинформационных технологий для оперативного реагирования на событие. Рассмотрены возможности аппроксимации сложных функций по методу Дж. Зойтендейка. При создании алгоритма использован метод возможных направлений в решении задач чебышевского приближения с дополнительными ограничениями.
Ключевые слова: аппроксимация, алгоритм, функция, ограничение, вектор.

Не зважаючи на досить великий обсяг робіт з апроксимації складних функцій, задача точності таких обрахунків залишається актуальною. Ця актуальність наразі проявляється при роботі з програмами, що дозволяють візуалізувати за допомогою геоінформаційних технологій зони ураження сильнодіючими отруйними речовинами (СДОР). Слід зазначити, що інструменти моделювання дозволяють наносити ці зони на карти і схеми у вигляді кола, півкола або сектора, який має кутові розміри і радіус, рівний глибині зараження. Зона фактичного зараження, як правило, має форму еліпса включається у зону можливого зараження. Така візуалізація не дає картини, що є наближеною до реальності, адже є різні особливості місцевості і хмара СДОР не буде чітким еліпсом чи колом. Крім того, у разі хімічних аварій є багато речовин, які можуть розповсюджуватися зваженою хмарою, яка заповнюватиме певні нерівності території (наприклад, яри) або вкриватиме річкові схили,

зависатиме з проміжком від поверхні землі. Такі випадки можна прорахувати, враховуючи особливості речовин, напрями розповсюдження і пересіченість місцевості, проте дуже важко представити у вигляді візуальної моделі для осіб, що приймають рішення (ОПР). Також візуалізація розповсюдження хмари ЗДОР не може бути представлена у вигляді кола чи сектора і у випадку, коли на шляху хмари буде річка і хмару СДОР частково потягне за течією. Високі будівлі і споруди на шляху СДОР також частково розірвуть контур.

Огляд існуючих моделей лінійного програмування, які здебільшого використовуються для прорахунків ситуацій, що зазначені, доводить, що ці моделі не завжди адекватні реальним ситуаціям. Наприклад, застосування широко розповсюджених градієнтних методів може бути неефективним в задачах яружної цільової функції, тобто, коли лінії цільової функції сильно витягнуті (мають форму еліпсів) у межах оптимальної точки. Такі функції є складними і дослідники уникають їх використання – «дно» таких функцій може бути прямим або звивистим і представляє собою підмножину точок, де поділ на існуючі і неіснуючі змінні зникає, тож з будь-якого напрямку функція змінюється повільно.

Метою роботи є представлення алгоритму опису яружних цільових функцій.

Слід зазначити, що питання апроксимації функцій свого часу широко були досліджені Василенко В.О. [1], Дзядиком В.К. [2], Поповим Б.О. [3], Люком Ю. [4] та іншими українськими і зарубіжними вченими. Проте в роботі використовуються елементи підходу, запропоновані Дж. Зойтендейком [5], що був представлений ще у 1963 році, проте розглядався дуже мало з причин важкості розрахунків за цим методом. У наш час, коли навіть персональні комп'ютери дозволяють вирішувати складні задачі, цей метод є цікавим для подальших досліджень та застосування в практичній роботі.

Для представлення алгоритму стисло розглянемо задачу апроксимації деякої функції.

Нехай на проміжку $[a,b]$ задана безперервна обмежена функція $f(x)$. Розглянемо кусочно-поліноміальну функцію $P(x)\in C^1(a,b)$, яка найкращим чином наближує $f(x)$за підходом Чебишева. Виразом $C^1(a,b)$ означимо клас функцій, безперервних на відрізку $[a,b]$ разом з першою похідною. Явно, що для $P(x)$ матиме місце наступний вираз:

$$\begin{cases} f_1(x) & x\in[a,C_1] \\ f_2(x) & x\in[C_1,C_2] \\ \dots\dots\dots\dots\dots\dots\dots\dots & \\ f_S(x) & x\in[C_S,b] \end{cases} \qquad (1)$$

Точки $a=C_0<C_1<C_2<...<C_S<C_{S+1}=b$ приймемо як невідомі.

Функції $f_i(x)$, $i=\overline{1,S}$ є поліноміальними зі ступенем не менше 2. Тобто, наведена задача, у випадку якщо $f_i(x)$ має однаковий ступінь, і є задачею побудови сплайн функції з фіксованими вузлами.

Задача побудови $P(x)$ зводиться до кількох завдань представлення поліномів найкращого наближення $f_i(x)$ в розумінні підходу Чебишева до функції $f(x)$ для $x\in[C_i,\, C_{i+1}]\,(1=\overline{0,k})$. Цей факт виходить з принципу оптимальності Беллмана. Саме

тому достатньо розглянути задачу побудови полінома найкращого наближення до $f(x)$ на деякому інтервалі. Цей поліном повинен ще задовольняти умови, що забезпечують відповідну гладкість $P(x)$.

Нехай задана функція $f(t)$ і деяка дискретна множина точок:

$$\mathrm{E}=\{Y_0, Y_1 ..., Y_{N+1}\} \in [a,b]$$

$$Y_0 = a, \quad Y_{N+1} = b.$$

Треба відшукати поліном заданого ступеня k:

$$\Pi_k(t)=\sum_{i=0}^{k} x_i t^i .$$

який мінімізує величину $\varepsilon(x)=\max_{t_i \in \mathrm{E}} |f(t_i)-\Pi_k(t_i)|$ по усіх x з області $\Delta \subset \mathrm{E}_{n+1}$, де Δ визначається:

$$\Delta = \{x \in \mathrm{E}_{n+1} : f^{(i)}(a)=\Pi_k^{(i)}(a), f^{(i)}(b)=\Pi_k^{(i)}(b), i=0,1\}.$$

Якщо прийняти, що $t_j^i = a_{ij} \left| i=\overline{0,k};\ j=\overline{0,n+1} \right|$, то задача, що розглядається, буде еквівалентною наступній задачі лінійного програмування:

$$\begin{cases} \min \varepsilon \\ \sum_{i=0}^{k} a_{ij} x_i - \varepsilon \le f(Y_i) \\ -\sum_{i=0}^{k} a_{ij} x_i - \varepsilon \le -f(Y_i) \\ \sum_{i=0}^{k} a_{i,0} x_i = f(Y_0) \\ \sum_{i=0}^{k} a_{i,n+1} x_i = f(Y_{n+1}) \\ \sum_{i=1}^{k} i \cdot a_{i-1,0} x_i = f'(Y_0) \\ \sum_{i=1}^{k} i \cdot a_{i-1,n+1} x_i = f'(Y_{n+1}) \\ \varepsilon \ge 0;\ j=\overline{1,n} \end{cases} \qquad (2)$$

Специфіка наведеної задачі лінійного програмування полягає у тому, що матриця обмежень $A=(a_{ij})\ i=0,k;\ j=0,n+1$ має прямокутний вигляд і кількість рядків домінує над кількістю стовпців, $K << N$. Тому для вирішення поставленої задачі обрано метод можливих напрямків Дж. Зойтендейка [5]. Через те, що метод передбачає наявність нерівності, то умова виразу (2) буде розписана як дві нерівності і на майбутнє буде припущено, що всі обмеження (2) мають вигляд нерівностей.

Тепер безпосередньо звернемося до методу можливих напрямів Дж. Зойтендейка та представимо алгоритм вирішення поставленої задачі.

Нехай нам дана довільна задача лінійного програмування:

$$\begin{cases} \max \sum_{j=1}^{k} d_j x_j \\ \sum_{j=1}^{k} a_{ij} x_j \le b_i \\ x_j \ge 0,\ i = \overline{1,P};\ j = \overline{1,k} \end{cases} \qquad (3)$$

Як і всі методи лінійного програмування, градієнтний метод вимагає відшукання точки, яка задовольняє обмеження задачі лінійного програмування. Позначимо її $X^0 = \left(x_1^0,...,x_k^0\right)$. Тоді для X^0 виконується:

$$\sum_{j=1}^{k} a_{ij} x_j^0 \le b_i; \qquad x_j^0 \ge 0 \quad i = \overline{1,P};\ j = \overline{1,k}. \qquad (4)$$

На відміну від симплексного і двоїстого методів, вирішення задачі лінійного програмування X^0 може й не бути базисною точкою, що значно спрощує вирішення задачі. У цьому дослідженні припустимо, що така точка нам відома. Тоді ірраціональна процедура знаходження рішення задачі (3) зводиться до наступного: з точки X^0 обираємо напрямок S, за яким величина $\sum_{j=1}^{k} d_j S_j$ має найбільше значення і вектор $S = \left(S_1,...,S_k\right)$ задовольняє обмеження $\sum_{j=1}^{k} P_{ij} S_j \le 0,\ i = \overline{1,P_1}\ \left(P_1 \le P + K\right)$, де матриця $P = \left(P_{ij}\right)$ складена з умов матриці обмежень (3), які для точки X^0 виконуються як рівняння, тобто, для матриці P маємо:

$$\sum_{j=1}^{k} P_{ij} x_j^0 = b_1 \qquad i = \overline{1,P_1},$$

Додаючи сюди і умову невід'ємного невідомого. Після обрання напряму S, обираємо довжину кроку λ для переходу у наступну точку X^1, виходячи з умови, що X^1 повинна задовольняти (4). Точка, на якій зупиниться процес, буде вирішенням задачі (3).

Для побудови алгоритму слід більш детально зупинитися на виборі напрямку S. Знаходження вектора $S = \left(S_1,...,S_k\right)$ зводиться до знаходження рішення наступної задачі математичного програмування:

$$\sum_{j=1}^{k} d_j S_j \to \max \qquad (5)$$

$$\sum_{j=1}^{k} P_{ij} S_j \le 0 \qquad \left(i = \overline{1,P_1}\right)$$

до якої, як правило, додають ще одне обмеження (нормалізацію) на вектор $S = \left(S_1,...,S_k\right)$. Для дослідження обираємо обмеження:

$$\sum_{j=1}^{k} S_j^2 \le 1. \qquad (7)$$

Оскільки застосовуючи запропонований метод на практиці, дуже важко буде вибрати деяку точку X^0, яка задовольнятиме (4), то замість задачі (3) можна

вирішити задачу, яка у деякому сенсі є еквівалентною задачі (3), тобто застосувати метод можливих напрямків до вирішення задач чебишевського наближення з додатковими обмеженнями:

$$\begin{cases} \sum_{j=1}^{k} d_j x_j - M_\xi \to \max \\ \sum_{j=1}^{k} a_{ij} x_j + \beta_i \xi \le b_i \qquad i = \overline{1,P} \\ x_j \ge 0, \quad \xi \ge 0 \qquad j = 1,K \end{cases} \tag{8}$$

де M є великим невід'ємним числом, а величини визначаються системою

$$\beta_i = \begin{cases} 0, \ \textit{якщо} \ \ b_i \ge 0 \\ -1, \ \textit{якщо} \ \ b_i < 0, \ \ i = \overline{1,P} \end{cases}.$$

Припустімо, значення невідомої ξ дорівнює $\xi_0 = \{\max(-b_i)/b_i < 0, \quad i = \overline{1,P}\}$. У зазначеному випадку вектор $X_\xi^0 = (O_1 O, ... O_1 \xi_0)$ стане початковим вирішенням задачі (8). А якщо область умов, що задана у (4), є не порожньою, то задача (8) матиме оптимальне рішення, а невідома ξ дорівнюватиме 0. Саме тому, у разі отримання від'ємного рішення задачі (8) $X_\xi^{on} = \{X_1^{on}, X_2^{on}, ..., X_k^{on}, O\}$ ми матимемо і оптимальне рішення задачі (3) $X_{on} = \{X_1^{on}, X_2^{on}, ..., X_k^{on}\}$.

Враховуючи, що ентропія інформації описується моделлю, яка визначає невизначеність повної групи випадкових подій або випадкових станів $E = -\sum_{i=1}^{n} p_i \log p_i$, а за змістом представляє собою зворотну величину до кількості інформації, можлива деяка кількість n випадкових подій з імовірністю $p_1 ... p_n$, які не відповідатимуть умовам, що прийняті. Наприклад, число M буде числом з плаваючою точкою, коли неможливо представити нуль для невідомої ξ. Враховуючи, що при розрахунках на обчислювальній техніці отримання нуля залежить від багатьох факторів, то для побудови початкового вирішення задачі (2) можна застосувати додаткові перетворення і розглянути наступну задачу (9). Перехід від задачі (2) до задачі (9) обумовлюється тим, що значення точок t_i та $f(t_i)$, а також обчислення похідних $f'(a)$, $f'(b)$, завжди мають деяку похибку. Саме тому, замість рівностей (2), можна обмежитися вимогами виконання відповідних умов нерівностей:

$$|\Pi_k(a) - f(a)| \le \alpha_1 \varepsilon$$

$$|\Pi'_k(a) - f'(a)| \le \alpha_2 \varepsilon$$

Аналогічно будуються умови для точки b. Величини α_1, α_2 додатні та обрані в залежності від необхідної точності виконання нерівностей.

$$\begin{cases} -\varepsilon \to \max \\ \sum_{i=0}^{k} a_{ij} x_i - \varepsilon \le f(Y_j) \\ -\sum_{i=0}^{k} a_{ij} x_i - \varepsilon \le -f(Y_j) \\ \sum_{i=0}^{k} a_{i,0} x_i - \alpha_1 \varepsilon \le f(Y_0) \\ -\sum_{i=0}^{k} a_{i,0} x_i - \alpha_1 \varepsilon \le -f(Y_0) \\ \sum_{i=0}^{k} a_{i,N+1} x_i - \alpha_1 \varepsilon \le f(Y_{N+1}) \\ -\sum_{i=0}^{k} a_{i,N+1} x_i - \alpha_1 \varepsilon \le -f(Y_{N+1}) \\ \sum_{i=1}^{k} i \cdot a_{i-1,0} x_i - \alpha_2 \varepsilon \le f'(Y_0) \\ -\sum_{i=1}^{k} i \cdot a_{i-1,0} x_i - \alpha_2 \varepsilon \le -f'(Y_0) \\ \sum_{i=1}^{k} i \cdot a_{i-1,N+1} x_i - \alpha_2 \varepsilon \le f'(Y_{N+1}) \\ -\sum_{i=1}^{k} i \cdot a_{i-1,N+1} x_i - \alpha_2 \varepsilon \le -f'(Y_{N+1}) \\ \varepsilon \ge 0, \quad j = 1, 2, ..., N \end{cases} \tag{9}$$

Вибір початкового вирішення для системи (9) можна отримати наступним чином.

Візьмемо $x_i = 0,\ i = O, K$. Значення ε визначаємо за формулою

$$\varepsilon^0 \left\{ \max \frac{-f(\mathrm{Y}_i)}{a_{i,\varepsilon}} \middle| f(\mathrm{Y}_i) < 0, \quad i = \overline{O, N+1} \right\},$$

де $a_{i,\varepsilon}$ – коефіцієнт при ε $(a_{i,\varepsilon} = -1, -\alpha_1, -\alpha_2)$.

У цьому випадку точка $X^0 = \left\{ \underbrace{O_1 O, ... O}_{k+1} \ \ \varepsilon^0 \right\}$ буде задовольняти обмеженням задачі (9), тобто, її можна означити як початкову точку.

Таким чином, вирішуючи задачу (9) методом можливих напрямків, можна отримати вирішення задачі найкращого наближення поліномом $\Pi_k(x)$ функції $f(x)$ на $[a,b]$

Далі припустимо, що поставлену задачу слід вирішити на кінцевому проміжку $[\alpha, \beta]$, де функція $f(x)$, що апроксимується, є обмеженою, безперервною та

однозначною. При обраному масштабі оберемо відрізок одиничної довжини $[C_0, C_1]$. Також приймаємо, що всередині цього відрізку відсутні полюсні точки. Попередньо досліджуємо на цьому відрізку $f(x)$. Приймаємо, що $C_1 > C_0$. Нехай відрізок розбито на N частин, тобто $N+1$ точки апроксимації є заданими. Серед них є:

а) точки C_0 і C_1;

б) $\overline{N}$ фіксованих точок;

в) $(N+1) - \overline{N} - 2 = N - \overline{N} - 1$ звичайних точок.

Нехай x_j^0 - довільна точка ділення; $x_0^0 = C_0$ і $x_{N+1}^0 = C_1$. Вважаємо, що нам задані значення $f(x)$ в кожній із точок x_j^0, тобто задана сукупність $f(x_j^0)$.

Визначимо

$$\delta_{1,j}^{(0)} = f(x_{j+1}^0) - f(x_j^0), \tag{10}$$

а також обчислюємо величини

$$\sigma_{1,j}^{(0)} = \frac{\delta_{1,j}^{(0)}}{x_{j+1}^0 - x_j^0} \ . \tag{11}$$

Тепер досліджуємо поведінку функції $\sigma_1^{(0)}(x)$, заданої точками $\sigma_{1,j}^{(0)}$ на відрізку $[C_0, C_1]$. Будемо розрізняти наступні три випадки:

1) $\left|\sigma_{1,j+1}^{(0)} - \sigma_{1,j}^{(0)}\right| \le \xi^{(0)}$, де $\xi^{(0)} > 0$ – досить мале число. Тобто, функція $\sigma_1^{(0)}(x)$ з відомою точністю поводиться як постійна величина. У цьому випадку природно покласти ступінь полінома рівній одиниці $n = 1$.

2) Функція $\sigma_1^{(0)}(x)$ – знакопостійна на $[C_0, C_1]$ й $\sigma_{1,j}^{(0)} \ne 0$. Тоді $n = 2$. Якщо ж $\sigma_1^{(0)}(x)$ приймає нульове значення в m різних точках, між якими перебувають $\sigma_{1,j}^{(0)}$ відмінні від нуля, то $n = m + 2$.

3) $\sigma_1^{(0)}(x)$ – знакозмінна на $[C_0, C_1]$. З врахуванням того, що число N настільки велике щодо довжини відрізка, що ймовірність втрати зміни знака стає дуже малою. Нехай $\sigma_1^{(0)}(x)$ змінює знак у l точках з $[C_0, C_1]$ і крім цього в m різних точках $\sigma_{1,j}^{(0)} = 0$ (між цими точками обов'язково перебувають такі, у яких $\sigma_{1,j}^{(0)} \ne 0$), але не має знака в їх оточенні. Нехай далі в P точках, зазначених $l + m \,/\, f(x_j^0) = 0$. Тоді ступінь полінома

$$n = l + m + p + 1 \tag{12}$$

Задамося деяким числом $\tilde{\varepsilon} > 0$. Будемо говорити, що $P(x)$_задовольняє по точності, якщо

$$\max_{x \in [a,b]} \left|P(x) - f(x)\right| \le \tilde{\varepsilon}$$

Очевидно, що кожний поліном $f_i(x)$, що становить $P(x)$, повинен задовольняти по точності.

Нехай на деякому інтервалі $[\alpha, \beta] \subset [a,b]$ побудовано поліном $f_s(x)$ ступеня n Побудувавши відповідну задачу лінійного програмування, як було показано вище (9)

і вирішивши її, ми одержимо величину ε. Вирішення задачі відшукання $f_s(x)$ з уточненням ступеня полінома й підінтервалу апроксимації є кінцевою.

Враховуючи викладене, можна визначити перелік фактичних параметрів, які потрібні для використання в алгоритмі, для створення програми опису яружних цільових функцій:

$P1$ – кількість полюсних точок на $[a,b]$, включаючи й кінці інтервалу апроксимації.

$P2$ – кількість фіксованих точок на $[a,b]$.

NO – кількість точок на відкритому інтервалі $(O,E1)$.

$\Phi1$ – кількість пар $(x_i, f(x_i))$, які задають таблицю функції $f(x)$, $x \in [a,b]$.

$E1$ – довжина підінтервалів апроксимації (попередній крок перебирання $[a,b]$).

Якщо виконується протилежна нерівність, то число підінтервалів апроксимації буде більшим, ніж це передбачено вектором полюсних точок $C[1:P1]$. Цим будуть уведені нові полюсні точки.

$D2$ – задана точність апроксимації (відповідає $\tilde{\varepsilon}$).

$D5$ – відповідає величині α_2

$D6$ – відповідає величині α_1

$C[1:P1]$ – масив полюсних точок, записаних у порядку зростання, включає точки a й b.

$Z[1:P1]$ – масив фіксованих точок, записаних у порядку зростання.

$Y[1:NO]$ – масив точок з відкритого інтервалу $(O,E1)$.

$XX, FF[1:\Phi1]$ – масиви відповідних точок

$x_j \in [a,b]$ и $f(x_j)$ $j = 1,..., \Phi1$

$(f(XX[J]) = FF[J]$, $J = 1,..., \Phi1)$

Власне, алгоритму опису яружних цільових функцій, буде наступним:

1. Здійснюється введення необхідних параметрів. Лічильники H, KO, K і перший елемент масиву $XX[0]$ представляють первісні значення.
2. Вибір фактичних параметрів і визначення величин, потрібних для роботи програми.
3. Визначається величина D – довжина підінтервалу апроксимації.
4. Здійснюється одночасна побудова масивів X, E, F, $F1$.
5. Визначається попередній ступінь $N2$ полінома апроксимації за допомогою масивів X і F.
6. Здійснюється зменшення довжини D підінтервалу на половину $D: = D/2$ й відновлення лічильників H, K, KO.
7. Побудова матриці обмежень за допомогою масивів E, Z, $F1$.
8. Побудова $C1$ й початкове рішення $X1$ задачі лінійного програмування.
9. Побудова напрямку $S1$.
10. Обчислення довжини кроку для побудови нового вектора $X1$.
11. Побудова нового вектора $X1$.
12. Блок коректування ступеня.

13. Зниження ступеня. Побудова нових обмежень задачі лінійного програмування.
14. Підвищення ступеня. Побудова обмежень задачі лінійного програмування.
15. Фіксація результатів і перехід до нового підінтервалу апроксимації.

Таким чином, у наведеній роботі представлений первинний алгоритм опису «яружних» цільових функцій, достатній для програмної реалізації мовами програмування або за допомогою мови опису алгоритмів UML.

Перспективою представленого дослідження є прорахунок конкретних ситуацій, які описуються яружними функціями та візуалізація результатів.

Враховуючи те, що задачі отримання рівномірних наближень сплайнами з мінімальною погрішністю розвивалися у багатьох роботах лише у теоретичному плані, практична розробка з метою програмної реалізації є актуальною та необхідною. Слід зазначити, що наведений підхід та алгоритм може бути застосований у сфері підтримки прийняття рішень для вирішення багатьох задач, пов'язаних з описом складних об'єктів, розробки програм для пожежних роботів, що призначені входити у закриті приміщення для виконання своїх функцій і задач, працювати на територіях радіаційного забруднення та виконання інших завдань з візуалізації ситуації за допомогою геоінформаційних технологій.

Література

1. Василенко В.А. Сплайн-функции: теория, алгоритмы, программы. – Новосибирск: Наука, 1983. – 218 с.
2. Дзядик В.К. Введение в теорию равномерного приближения функций полиномами. – М.: Наука, 1977. – 512 с.
3. Попов Б.А. Равномерное приближение сплайнами. – К.: Наук. думка, 1989. – 272 с.
4. Люк Ю. Специальные математические функции и их аппроксимации. – М.: Мир, 1980. – 608 с.
5. Зойтендейк Г. Методы возможных направлений. – М.: Издательство Иностранной литературы, 1963. – 178 с.

Literatura

1. Vasilenko V.A. Spline functions: theory, algorithms, programs. – Novosibirsk: Sciense, 1983. – 218 p.
2. Dzjadyk V.K. Introduction to the theory of uniform approximation of functions by polynomials – M.: Sciense, 1977. – 512 p.
3. Popov B.A. Uniform approximation by splines. – Kyiv: Publishing house "Scientific idea" of NASU, 1989. – 272 p.
4. Luck Y. Mathematical functions and their approximations. – M.: World, 1980. – 608 p.
5. Zoutendijk G. Methods of Feasible Directions. – M.: Foreign Literature Publishing House, 1963. – 178 p.

RESUME

O.M. Trofymchuk, O.O. Kryazhych

Algorithm descriptions gully objective functions

The algorithm descriptions gully objective functions. This algorithm can be used for detailed visualization of the distribution of zones of defeat of highly toxic substances in the event of industrial disaster. This rendering is required when using geoinformation technologies for rapid response to an event.

The use of gradient methods may not be effective in describing functions gully. These are functions which have greatly elongated lines (ellipses) within the boundaries of the optimum point. When creating the algorithm proposed use approximation of complex functions by the method's G. Zoutendijk.

This paper presents the initial algorithm. This algorithm can be implemented in programming languages or graphical description language UML. When creating the algorithm used method possible directions in problem solving Chebyshev's approximation restrictions.

Calculation of real situations on this algorithm is a promising study.

О.М. Трофимчук, О.О. Кряжич

Алгоритм опису яружних цільових функцій

У статті представлений алгоритм опису яружних цільових функцій, який пропонується використовувати для детальної візуалізації розповсюдження зон ураження сильнодіючих отруйних речовин у разі виникнення техногенної аварії. Така візуалізація є необхідною і можливою при застосуванні геоінформаційних технологій для оперативного реагування на подію.

Застосування широко розповсюджених градієнтних методів може бути неефективним в задачах яружної цільової функції, тобто, коли лінії цільової функції сильно витягнуті і мають форму еліпсів у межах оптимальної точки. Саме тому, в роботі при побудові алгоритму пропонується використання підходу з апроксимації складних функцій за методом Дж. Зойтендейка.

У наведеній роботі представлений первинний алгоритм опису «яружних» цільових функцій, достатній для програмної реалізації мовами програмування або за допомогою мови опису алгоритмів UML. При побудові алгоритму застосований метод можливих напрямків до вирішення задач чебишевського наближення з додатковими обмеженнями.

Перспективою представленого дослідження є прорахунок конкретних ситуацій, які описуються яружними функціями та візуалізація результатів.

Надійшла до редакції 01.07.2015

